

Listing of Claims:

1-2. (Canceled)

3. (Currently Amended) A method for automated tool management comprising the steps of:
~~The method as recited in Claim 1, wherein if said first request comprises a request for data and if~~
~~said first tool is an asynchronous source of said data, then the method further comprises the steps~~
~~of:~~

receiving a first message in a first selected protocol from a first client application,
wherein said first message comprises a first request to perform a first action on a first
tool;

extracting a first pointer from the contents of said first message in order to identify a first
object in an equipment model, wherein said equipment model comprises a logical
representation of said first tool;

invoking a first procedure of said first object in response to said first message;

transferring a first return value to said first client application, wherein said first return
value is associated with said first action;

receiving a second message in a second selected protocol from a second client
application, wherein said second message comprises a second request to perform a
second action on said first tool;

extracting a second pointer from the contents of said second message in order to identify
said first object in said equipment model, wherein said second selected protocol is
different than said first selected protocol;

invoking a second procedure of said first object in response to said second message; and

transferring a second return value to said second client application, wherein said second
return value is associated with said second action;

wherein if said first request comprises a request for data and if said first tool is an
asynchronous source of said data, then the method further comprises the steps of:

if valid information exists corresponding to said data, creating said first return
value based on said valid information;

if valid information does not exist corresponding to said data, creating said first return value based on a database of said equipment model;
incorporating said first return value into a return message to said first client application; and
transferring said return message in said selected protocol to said client application in response to an address provided by said client application.

4-7. (Canceled)

8. (Currently Amended) The method as recited in Claim 3 ~~4~~, wherein said first protocol and said second protocol comprise protocols selected from the following: Component Object Model (COM), Remote Method Invocation (RMI), CORBA, Simple Object Access Protocol (SOAP), SECS, GEM, HyperText Markup Language (HTML), Extensible Markup Language (XML).

9. (Currently Amended) The method as recited in Claim 3 ~~4~~, wherein said first procedure of said first object is invoked to remotely access and electronically diagnose said first tool.

10-12. (Canceled)

13. (Currently Amended) A computer program product having a computer readable storage medium having computer program logic recorded thereon for automated tool management, comprising: ~~The computer program product as recited in Claim 11, wherein if said first request comprises a request for data and if said first tool is an asynchronous source of said data, then the computer program product further comprises:~~

programming operable for receiving a first message in a first selected protocol from a first client application, wherein said first message comprises a first request to perform a first action on a first tool;

programming operable for extracting a first pointer from the contents of said first message to identify a first object in an equipment model, wherein said equipment model comprises a logical representation of said first tool;

programming operable for invoking a first procedure of said first object in response to said first message;

programming operable for transferring a first return value to said first client application,
wherein said first return value is associated with said first action;

programming operable for receiving a second message in a second selected protocol from
a second client application, wherein said second message comprises a second request
to perform a second action on said first tool;

programming operable for extracting a second pointer from the contents of said second
message to identify said first object in said equipment model, wherein said second
selected protocol is different than said first selected protocol;

programming operable for invoking a second procedure of said first object in response to
said second message;

programming operable for transferring a second return value to said second client
application, wherein said second return value is associated with said second action;

wherein if said first request comprises a request for data and if said first tool is an
asynchronous source of said data, then the computer program product further
comprises:

if valid information exists corresponding to said data, programming operable for
creating said first return value based on said valid information;

if valid information does not exist corresponding to said data, programming
operable for creating said first return value based on a database of said
equipment model;

programming operable for incorporating said first return value into a return
message to said first client application; and

programming operable for transferring said return message in said selected
protocol to said client application in response to an address provided by said
client application.

14-17. (Canceled)

18. (Currently Amended) The computer program product as recited in Claim 13 44, wherein
said first protocol and said second protocol comprise protocols selected from the following:

Component Object Model (COM), Remote Method Invocation (RMI), CORBA, Simple Object Access Protocol (SOAP), SECS, GEM, HyperText Markup Language (HTML), Extensible Markup Language (XML).

19. (Currently Amended) The computer program product as recited in Claim ~~13~~ 44, wherein said first procedure of said first object is invoked to remotely access and electronically diagnose said first tool.

20-22. (Canceled)

23. (Currently Amended) A system, comprising: a processor; a memory unit storing a computer program operable for storing a computer program operable for automated tool management; and a bus system coupling the processor to the memory, wherein the computer program is operable for performing the following programming steps: The system as recited in Claim 21, wherein if said first request comprises a request for data and if said first tool is an asynchronous source of said data, then the computer program is further operable for performing the following programming steps:

receiving a first message in a first selected protocol from a first client application, wherein said first message comprises a first request to perform a first action on a first tool;

extracting a first pointer from said first message to identify a first object in an equipment model, wherein said equipment model comprises a logical representation of said first tool;

invoking a first procedure of said first object in response to said first message;

transferring a first return value to said first client application, wherein said first return value is associated with said first action;

receiving a second message in a second selected protocol from a second client application, wherein said second message comprises a second request to perform a second action on said first tool;

extracting a second pointer from said second message to identify said first object in said equipment model, wherein said second selected protocol is different than said first selected protocol;

invoking a second procedure of said first object in response to said second message;
transferring a second return value to said second client application, wherein said second
return value is associated with said second action;
wherein if said first request comprises a request for data and if said first tool is an
asynchronous source of said data, then the computer program is further operable for
performing the following programming steps:

if valid information exists corresponding to said data, creating said first return value based on said valid information;

if valid information does not exist corresponding to said data, creating said first return value based on a database of said equipment model;

incorporating said first return value into a return message to said first client application; and

transferring said return message in said selected protocol to said client application in response to an address provided by said client application.

24-27. (Canceled)

28. (Currently Amended) The system as recited in Claim 23 ~~24~~, wherein said first protocol and said second protocol comprise protocols selected from the following: Component Object Model (COM), Remote Method Invocation (RMI), CORBA, Simple Object Access Protocol (SOAP), SECS, GEM, HyperText Markup Language (HTML), Extensible Markup Language (XML).

29. (Currently Amended) The system as recited in Claim 23 ~~24~~, wherein said first procedure ~~method~~ of said first object is invoked to remotely access and electronically diagnose said first tool.

30. (Canceled)

31. (Currently Amended) The method as recited in Claim 3 ~~4~~ further comprising the steps of:
generating a security wrapper layer, wherein said security wrapper layer provides a layer of protection to said equipment model; and

creating a security wrapper object in said security wrapper layer, wherein a pointer to a corresponding equipment model object is stored in said security wrapper object.

32. (Previously Presented) The method as recited in Claim 31, wherein if said corresponding equipment model object is said object corresponding to said first request then a pointer to said corresponding security wrapper object is transferred to said first client application.

33. (Previously Presented) The method as recited in Claim 32 further comprising the step of: determining if said first selected action on said first tool can be performed in response to access rules stored in said corresponding security wrapper object.

34. (Previously Presented) The method as recited in Claim 33, wherein if said first selected action on said first tool can be performed then the method further comprises the step of: invoking a procedure by said corresponding security wrapper object to perform said first selected action.

35. (Currently Amended) The computer program product as recited in Claim 13 ~~14~~ further comprising:

programming operable for generating a security wrapper layer, wherein said security wrapper layer provides a layer of protection to said equipment model; and

programming operable for creating a security wrapper object in said security wrapper layer, wherein a pointer to a corresponding equipment model object is stored in said security wrapper object.

36. (Previously Presented) The computer program product as recited in Claim 35, wherein if said corresponding equipment model object is said first object corresponding to said first request then a pointer to said corresponding security wrapper object is transferred to said first client application.

37. (Previously Presented) The computer program product as recited in Claim 36 further comprises: programming operable for determining if said first selected action on said first tool

can be performed in response to access rules stored in said corresponding security wrapper object.

38. (Previously Presented) The computer program product as recited in Claim 37, wherein if said first selected action on said first tool can be performed then the computer program product further comprises: programming operable for invoking a procedure by said corresponding security wrapper object to perform said first requested action.

39. (Currently Amended) The system as recited in Claim ~~23~~ 24, wherein the computer program is further operable for performing the following programming steps:

generating a security wrapper layer, wherein said security wrapper layer provides a layer of protection to said equipment model; and

creating a security wrapper object in said security wrapper layer, wherein a pointer to a corresponding equipment model object is stored in said security wrapper object.

40. (Previously Presented) The system as recited in Claim 39, wherein if said corresponding equipment model object is said first object corresponding to said first request then a pointer to said corresponding security wrapper object is transferred to said first client application.

41. (Previously Presented) The system as recited in Claim 40, where the computer program is further operable for performing the following programming step: determining if said first selected action on said first tool can be performed in response to access rules stored in said corresponding security wrapper object.

42. (Previously Presented) The system as recited in Claim 41, wherein if said first selected action on said first tool can be performed then the computer program is further operable for performing the following programming step: invoking a procedure by said corresponding security wrapper object to perform said first selected action.

43. (Currently Amended) The method as recited in Claim ~~3~~ 4, wherein said step of receiving a first message and said step of transferring a first return value are performed by an application interface unit, wherein said application interface unit interfaces said first client application with said equipment model.

44. (Canceled)

45. (Currently Amended) The computer program product as recited in Claim 13 ~~44~~, wherein said programming operable for receiving a first message and said programming operable for transferring a first return value are implemented by an application interface unit, wherein said application interface unit interfaces said client application with said equipment model.

46. (Canceled)

47. (Currently Amended) The system as recited in Claim 23 ~~24~~, wherein said step of receiving a first message and said step of transferring a first return value are performed by an application interface unit, wherein said application interface unit interfaces said client application with said equipment model.

48-51. (Canceled)

52. (Currently Amended) A method for automated tool management comprising the steps of:
~~The method as recited in Claim 1, further comprising:~~

receiving a first message in a first selected protocol from a first client application,
wherein said first message comprises a first request to perform a first action on a first tool;

extracting a first pointer from the contents of said first message in order to identify a first
object in an equipment model, wherein said equipment model comprises a logical representation
of said first tool;

invoking a first procedure of said first object in response to said first message;

transferring a first return value to said first client application, wherein said first return
value is associated with said first action;

receiving a second message in a second selected protocol from a second client
application, wherein said second message comprises a second request to perform a second action
on said first tool;

extracting a second pointer from the contents of said second message in order to identify
said first object in said equipment model, wherein said second selected protocol is different than
said first selected protocol;

invoking a second procedure of said first object in response to said second message;

transferring a second return value to said second client application, wherein said second return value is associated with said second action;

receiving a third message in said first selected protocol from said first client application, wherein said third message comprises a third request to perform a third action on a second tool, wherein said third message identifies a second object in an equipment model, wherein said equipment model comprises a logical representation of said second tool;

invoking a third procedure of said second object in response to said third message; and

transferring a third return value to said first client application, wherein said third return value is associated with said third action.

53. (Currently Amended) A computer program product having a computer readable storage medium having computer program logic recorded thereon for automated tool management, comprising: The computer program product as recited in Claim 11, further comprising:

programming operable for receiving a first message in a first selected protocol from a first client application, wherein said first message comprises a first request to perform a first action on a first tool;

programming operable for extracting a first pointer from the contents of said first message to identify a first object in an equipment model, wherein said equipment model comprises a logical representation of said first tool;

programming operable for invoking a first procedure of said first object in response to said first message;

programming operable for transferring a first return value to said first client application, wherein said first return value is associated with said first action;

programming operable for receiving a second message in a second selected protocol from a second client application, wherein said second message comprises a second request to perform a second action on said first tool;

programming operable for extracting a second pointer from the contents of said second message to identify said first object in said equipment model, wherein said second selected protocol is different than said first selected protocol;

programming operable for invoking a second procedure of said first object in response to said second message;

programming operable for transferring a second return value to said second client application, wherein said second return value is associated with said second action;

programming operable for receiving a third message in said first selected protocol from said first client application, wherein said third message comprises a third request to perform a third action on a second tool, wherein said third message identifies a second object in an equipment model, wherein said equipment model comprises a logical representation of said second tool;

programming operable for invoking a third procedure of said second object in response to said third message; and

programming operable for transferring a third return value to said first client application, wherein said third return value is associated with said third action.

54. (Currently Amended) A system, comprising: a processor; a memory unit storing a computer program operable for storing a computer program operable for automated tool management; and a bus system coupling the processor to the memory, wherein the computer program is operable for performing the following programming steps: ~~The system as recited in Claim 21, wherein the computer program is further operable for performing the following programming steps:~~

receiving a first message in a first selected protocol from a first client application, wherein said first message comprises a first request to perform a first action on a first tool;

extracting a first pointer from said first message to identify a first object in an equipment model, wherein said equipment model comprises a logical representation of said first tool;

invoking a first procedure of said first object in response to said first message;

transferring a first return value to said first client application, wherein said first return value is associated with said first action;

receiving a second message in a second selected protocol from a second client application, wherein said second message comprises a second request to perform a second action on said first tool;

extracting a second pointer from said second message to identify said first object in said equipment model, wherein said second selected protocol is different than said first selected protocol;

invoking a second procedure of said first object in response to said second message;

transferring a second return value to said second client application, wherein said second return value is associated with said second action;

receiving a third message in said first selected protocol from said first client application, wherein said third message comprises a third request to perform a third action on a second tool, wherein said third message identifies a second object in an equipment model, wherein said equipment model comprises a logical representation of said second tool;

invoking a third procedure of said second object in response to said third message; and

transferring a third return value to said first client application, wherein said third return value is associated with said first action.